In the Claims:

Please amend claims 27, 29, 30, 32, 33 and 35 as indicated below.

1. (Original) A microprocessor, comprising:

an instruction cache configured to store instructions;

a branch prediction unit;

a trace cache configured to store a plurality of traces of instructions; and

a prefetch unit coupled to the instruction cache, the branch prediction unit, and the trace cache;

wherein the prefetch unit is configured to fetch instructions from the instruction cache until the branch prediction unit outputs a predicted target address; and

wherein if the prefetch unit identifies a match for the predicted target address in the trace cache, the prefetch unit is configured to fetch one or more of the plurality of traces from the trace cache.

2. (Original) The microprocessor of claim 1, wherein the branch prediction unit is configured to output the predicted target address in response to a prediction that a branch will be taken.

3. (Original) The microprocessor of claim 1, wherein the branch prediction unit is configured to output the predicted target address in response to detection of a branch misprediction.

4. (Original) The microprocessor of claim 1, further comprising a trace generator, wherein the trace generator is configured to begin a trace with an instruction corresponding to a label boundary.

5. (Original) The microprocessor of claim 4, wherein the trace generator is configured to check the trace cache for a duplicate copy of the trace that the trace generator is constructing.

6. (Original) The microprocessor of claim 5, wherein if the trace generator identifies a duplicate copy of the trace, the trace generator is configured to discard the trace under construction.

7. (Original) The microprocessor of claim 5, wherein if the trace generator identifies an entry corresponding to a duplicate copy of the trace, the trace generator is configured to check the trace cache for an entry corresponding to a next trace to be generated.

8. (Original) The microprocessor of claim 7, wherein if the trace generator identifies a trace entry corresponding to the next trace to be generated, the trace generator is configured to discard the trace under construction.

9. (Original) The microprocessor of claim 4, wherein the trace generator is configured to generate traces in response to instructions being retired.

10. (Original) The microprocessor of claim 4, wherein the trace generator is configured to generate traces in response to instructions being decoded.

11. (Original) The microprocessor of claim 1, wherein each of the plurality of traces comprises partially-decoded instructions.

12. (Original) The microprocessor of claim 1, wherein each of the plurality of traces is associated with a tag comprising the address of an earliest instruction, in program order, stored within that trace.

13. (Original) The microprocessor of claim 1, wherein each of the plurality of traces is associated with a flow control field comprising a label for an instruction to which control will pass for each branch operation comprised in that trace.

14. (Original) A computer system, comprising:

a system memory; and

a microprocessor coupled to the system memory, comprising:

an instruction cache configured to store instructions;

a branch prediction unit;

a trace cache configured to store a plurality of traces of instructions; and

a prefetch unit coupled to the instruction cache, the branch prediction unit, and the trace cache;

wherein the prefetch unit is configured to fetch instructions from the instruction cache until the branch prediction unit outputs a predicted target address; and

wherein if the prefetch unit identifies a match for the predicted target address in the trace cache, the prefetch unit is configured to fetch one or more of the plurality of traces from the trace cache.

15. (Original) The computer system of claim 14, wherein the branch prediction unit is configured to output the predicted target address in response to a prediction that a branch will be taken.

16. (Original) The computer system of claim 14, wherein the branch prediction unit is configured to output the predicted target address in response to detection of a branch misprediction.

17. (Original) The computer system of claim 14, further comprising a trace generator, wherein the trace generator is configured to begin a trace with an instruction corresponding to a label boundary.

18. (Original) The computer system of claim 17, wherein the trace generator is configured to check the trace cache for a duplicate copy of the trace that the trace generator is constructing.

19. (Original) The computer system of claim 18, wherein if the trace generator identifies a duplicate copy of the trace, the trace generator is configured to discard the trace under construction.

20. (Original) The computer system of claim 18, wherein if the trace generator identifies an entry corresponding to a duplicate copy of the trace, the trace generator is configured to check the trace cache for an entry corresponding to a next trace to be generated.

21. (Original) The computer system of claim 20, wherein if the trace generator identifies a trace entry corresponding to the next trace to be generated, the trace generator is configured to discard the trace under construction.

22. (Original) The computer system of claim 17, wherein the trace generator is configured to generate traces in response to instructions being retired.

23. (Original) The computer system of claim 17, wherein the trace generator is configured to generate traces in response to instructions being decoded.

24. (Original) The computer system of claim 14, wherein each of the plurality of traces comprises partially-decoded instructions.

25. (Original) The computer system of claim 14, wherein each of the plurality of traces is associated with a tag comprising the address of an earliest instruction, in program order, stored within that trace.

26. (Original) The computer system of claim 14, wherein each of the plurality of traces is associated with a flow control field comprising a label for an instruction to which control will pass for each branch operation comprised in that trace.

27. (Currently amended) A method, comprising:

receiving a retired instruction;

starting construction of a new trace if the received instruction is associated with a branch label;

if ~~the~~ a previous trace under construction duplicates a trace in a trace cache, delaying construction of the new trace until the received instruction corresponds to a branch label.

28. (Original) The method of claim 27, further comprising continuing construction of an incomplete trace already in process.

29. (Currently amended) The method of claim 27, further comprising searching the trace cache for duplicate entries.

30. (Currently amended) The method of claim 29, further comprising creating a new entry in the trace cache if no duplicate entry is identified.

31. (Original) The method of claim 29, further comprising discarding a trace if a duplicate entry is identified.

32. (Currently amended) A method, comprising:

fetching instructions from an instruction cache;

continuing to fetch instructions from the instruction cache until a branch target address is generated;

if a branch target address is generated, searching a trace cache for an entry corresponding to the branch target address.

33. (Currently amended) The method of claim 32, further comprising continuing to fetch instructions from the instruction cache if no entry is identified in the trace cache corresponding to the branch target address.

34. (Original) The method of claim 32, further comprising fetching one or more traces from the trace cache if an entry is identified in the trace cache corresponding to the branch target address.

35. (Currently amended) A microprocessor, comprising:

means for receiving a retired operation;

means for starting a new trace if the received operation is a first operation at a branch label;

means for delaying starting a new trace if ~~the~~ a previous trace under construction duplicates a trace in a trace cache, until the received operation corresponds to a branch label.